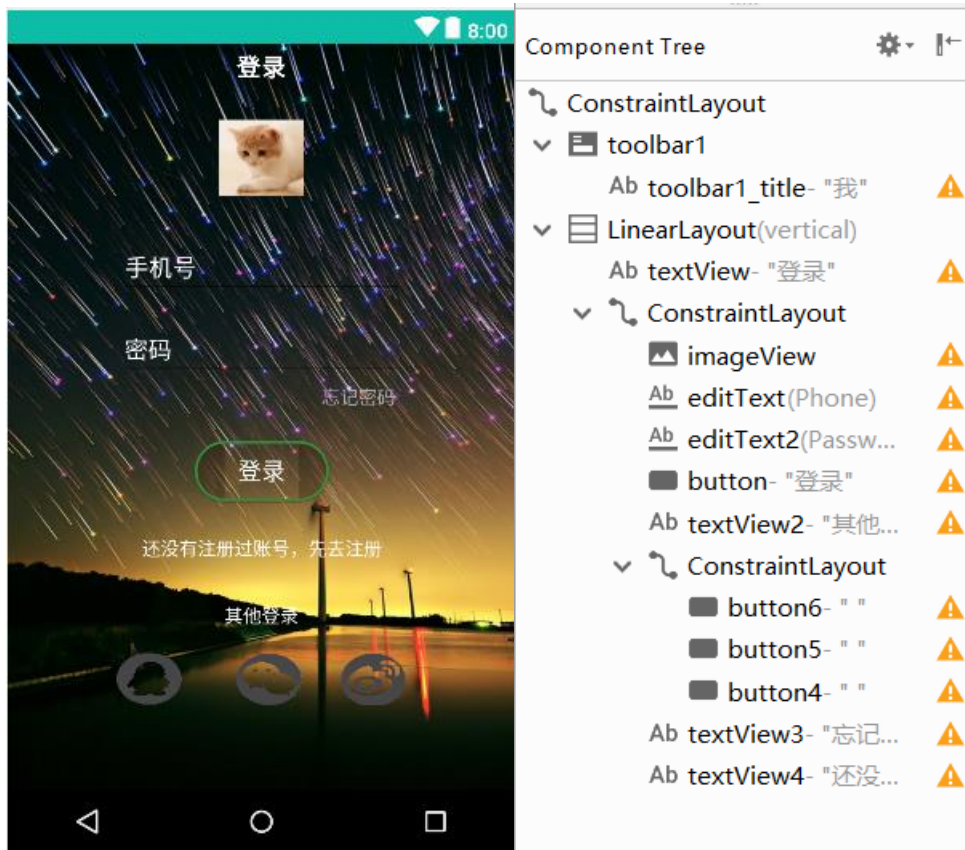


# 移动应用开发技术

## Homework 1

Android 应用界面 UI 设计，包括一个登录界面和一个注册界面：



登录界面包括标题、用户头像、用户 ID 编辑框、用户密码编辑框、忘记密码、登录按钮、跳转到注册界面链接、第三方登录按钮。

分别用到的控件包括：

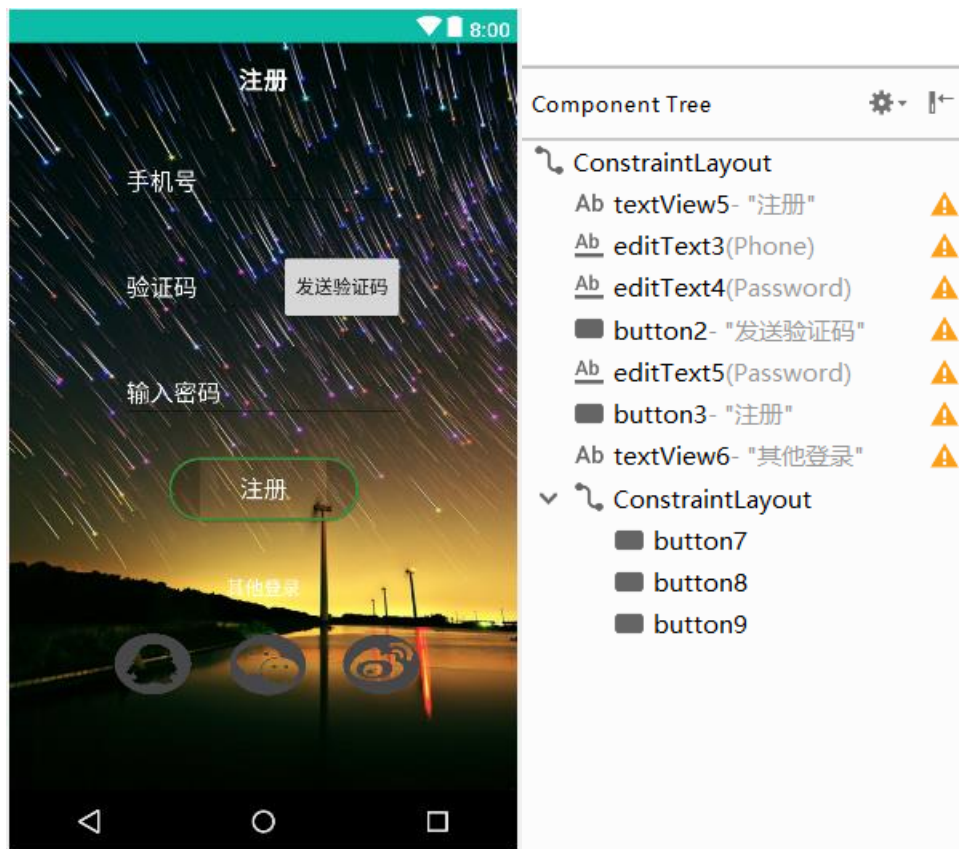
Layout: ConstraintLayout\LinearLayout

TextView

EditText

ImageView

Button



注册界面包括标题、手机号编辑框、发送验证码按钮、验证码编辑框、密码编辑框、注册按钮、第三方登录按钮。

分别用到的控件包括：

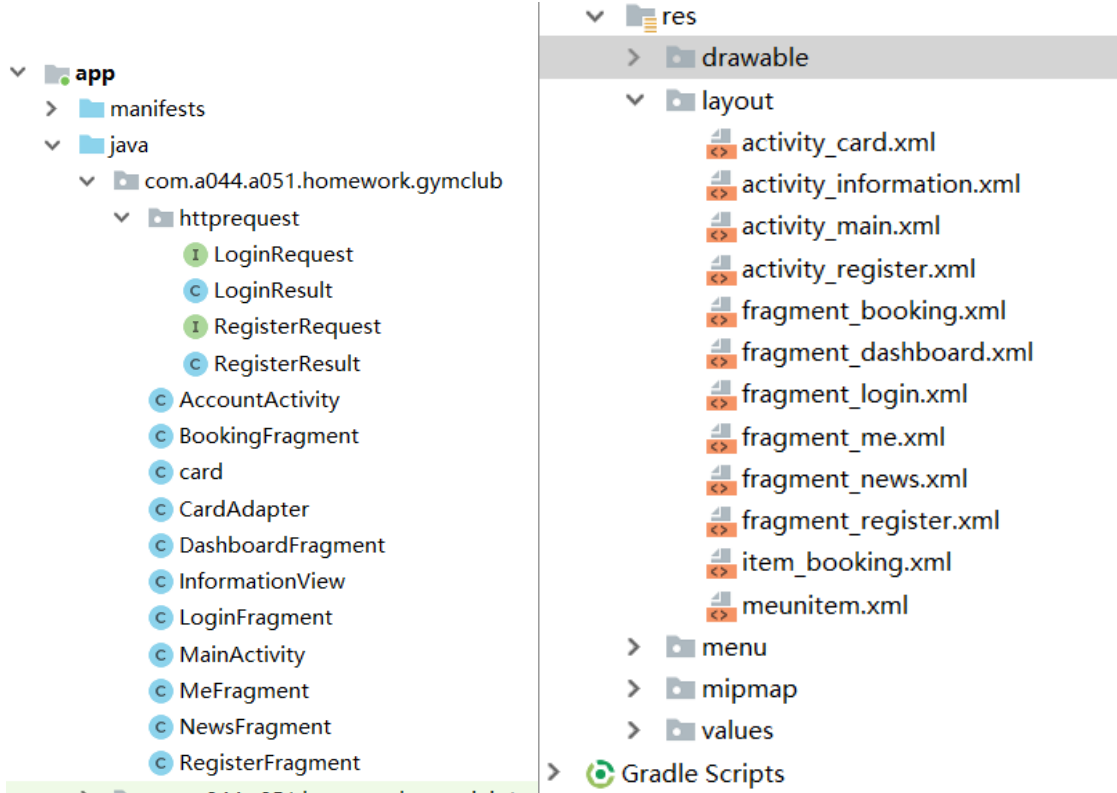
ConstraintLayout

TextView

EditText

Button

## Homework 2



实现了界面之间的跳转，加入了服务器实现了登录注册功能

## Homework 3

更加丰富了 Gymclub 应用的功能，运用了 VideoView 实现了视频的播放。

使用了 android.location 框架来实现了定位功能：

关键组件：

### LocationManager

这个组件并不需要实例化，通常是通过函数

`getSystemService(Context.LOCATION_SERVICE)`

获得，该函数会返回一个 LocationManager 的实例，例如：

```
LocationManager locationManager = (LocationManager)
this.getSystemService(Context.LOCATION_SERVICE);
```

### LocationListener

通过回调（callback）来获取用户定位信息，需要通过新建一个 LocationListener 类，并实现其中的几个回调方法

```
1 // Define a listener that responds to location updates
2 LocationListener locationManager = new LocationListener() {
3     public void onLocationChanged(Location location) {
4         // Called when a new location is found by the network location provider.
5         // 这个函数中的location就是获取到的位置信息
6     }
7
8     public void onStatusChanged(String provider, int status, Bundle extras) {}
9
10    public void onProviderEnabled(String provider) {}
11
12    public void onProviderDisabled(String provider) {}
13 };
```

### Location

这是一个数据类，保存了定位的信息。

## Homework 4

实现离线缓存功能

核心内容：

申请 sd 卡权限

既然要进行离线缓存就需要往手机本地存储写入内容，所以首先需要申请获得手机的“存储权限”然后才可以写入手机本地内容。

创建数据类，比如保存用户数据，创建 Person 类

```
public class Person implements Serializable {  
    String name;  
    String sex;  
    int age;  
}
```

编写保存 object 对象到文件，和读取缓存

## Homework 5

第三方登录功能的实现（或者说尝试。。）

以 QQ 登录为例

首先需要到腾讯开放平台进行应用接入的注册

选择 Android 平台就会获取 APPID 和 APPKEY 创建完成后我们需要到下载 SDKJar 包，将 jar 包添加到我们的项目中去

在 Androidmanifest 在里面加入权限和注册 Activity

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.shiran.qqlogindemo">

    <!-- QQ登录授权所需权限 -->
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <!-- 注册SDKActivity -->
        <activity
            android:name="com.tencent.tauth.AuthActivity"
            android:launchMode="singleTask"
            android:noHistory="true">
            <intent-filter>
                <action android:name="android.intent.action.VIEW" />
                <category android:name="android.intent.category.DEFAULT" />
                <category android:name="android.intent.category.BROWSABLE" />
                <data android:scheme="tencent1105602574" /> <!-- 开放平台获取的A
            </intent-filter>
        </activity>
        <activity android:name="com.tencent.connect.common.AssistActivity"
            android:theme="@android:style/Theme.Translucent.NoTitleBar"
            android:screenOrientation="portrait"/>
    </application>

</manifest>
```